

Numerics made easy: solving the Navier-Stokes equation for arbitrary channel cross-sections using Microsoft Excel

Citation for published version (APA):

Richter, C., Kotz, F., Giselbrecht, S., Helmer, D., & Rapp, B. E. (2016). Numerics made easy: solving the Navier-Stokes equation for arbitrary channel cross-sections using Microsoft Excel. *Biomedical Microdevices*, 18(3), [52]. <https://doi.org/10.1007/s10544-016-0070-2>

Document status and date:

Published: 01/06/2016

DOI:

[10.1007/s10544-016-0070-2](https://doi.org/10.1007/s10544-016-0070-2)

Document Version:

Publisher's PDF, also known as Version of record

Document license:

Taverne

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.umlib.nl/taverne-license

Take down policy

If you believe that this document breaches copyright please contact us at:

repository@maastrichtuniversity.nl

providing details and we will investigate your claim.

Numerics made easy: solving the Navier–Stokes equation for arbitrary channel cross-sections using Microsoft Excel

Christiane Richter¹ · Frederik Kotz¹ · Stefan Giselbrecht² · Dorothea Helmer¹ · Bastian E. Rapp¹

Published online: 28 May 2016
© Springer Science+Business Media New York 2016

Abstract The fluid mechanics of microfluidics is distinctively simpler than the fluid mechanics of macroscopic systems. In macroscopic systems effects such as non-laminar flow, convection, gravity etc. need to be accounted for all of which can usually be neglected in microfluidic systems. Still, there exists only a very limited selection of channel cross-sections for which the Navier–Stokes equation for pressure-driven Poiseuille flow can be solved analytically. From these equations, velocity profiles as well as flow rates can be calculated. However, whenever a cross-section is not highly symmetric (rectangular, elliptical or circular) the Navier–Stokes equation can usually not be solved analytically. In all of these cases, numerical methods are required. However, in many instances it is not necessary to turn to complex numerical solver packages for deriving, e.g., the velocity profile of a more complex microfluidic channel cross-section. In this paper, a simple spreadsheet analysis tool (here: Microsoft Excel) will be used to implement a simple numerical scheme which allows solving the Navier–Stokes equation for arbitrary channel cross-sections.

Keywords Numerics · Navier–Stokes · Microsoft Excel · Finite differences method · Velocity profile

1 Introduction

Fluid mechanics is notorious for its very complicated mathematics. The fundamental equation of fluid mechanics is the Navier–Stokes equation, a second-order inhomogeneous partial differential equation in vector form which still poses significant challenges for solvers even in the 21st century. Solving the Navier–Stokes equation yields the velocity field of a flow problem. In microfluidics we can simplify this equation significantly by exploiting the fact that for laminar, stationary and parallel flows several crucial terms of the equation can be dropped. However, even the resulting equation can only be solved analytically for a very limited choice of channel cross-sections. These equations, which allow the calculation of the velocity distribution and (from it) the flow rate, hydraulic resistance and similar important characteristics of the flow, are usually looked up in the reference literature for different channel cross-sections. The most commonly found cross-sections are rectangular (Waldbaur et al. 2013) (made e.g., by replication processes), circular or elliptical (Khodaparast et al. 2014; Wilhelm et al. 2013) (as in the case of capillaries, tubes, etc.) or triangular (Kim et al. 2016) (created by etching processes or laser ablation). However, whenever a microfluidic channel is not highly symmetric, there is a high chance that no analytical solution can be found at all. In all of these cases, numerical methods have to be used. There is a wide range of numerical packages available which can be used to solve the Navier–Stokes equation on irregular channel cross-sections. These packages are either based on finite volume methods (FVM) or, more commonly finite element methods (FEM).

Electronic supplementary material The online version of this article (doi:10.1007/s10544-016-0070-2) contains supplementary material, which is available to authorized users.

✉ Bastian E. Rapp
bastian.rapp@kit.edu

¹ Institute of Microstructure Technology (IMT), Karlsruhe Institute of Technology (KIT), Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

² Department of Complex Tissue Regeneration (CTR), MERLN Institute for Technology-Inspired Regenerative Medicine, Maastricht University, Maastricht, The Netherlands

However, it is often not required to turn to a commercial numerical solver to derive a solution to the Navier–Stokes equation for a more complex channel cross-section. In this paper we will demonstrate that a simple spreadsheet analysis tool such as Microsoft Excel can be used to accomplish this task within seconds. The spreadsheet developed implements a finite difference method (FDM) based numerical scheme with very high numerical stability and sufficient precision for most applications in microfluidics. The main advantage of using a spreadsheet tool (here: Microsoft Excel) over usage of a numerical solver package is ease of use and excellent accessibility. Furthermore, even users not accustomed to numerical solver packages are very likely to have little trouble using Microsoft Excel which is used in many laboratories for experimental data visualization, statistics and similar tasks.

2 Numerical scheme

2.1 Navier–Stokes equation for Poiseuille flow

The pressure-driven channel flow which we will discuss in this paper is generally referred to as Poiseuille flow. In vector form, the Navier–Stokes equation is given by

$$\rho \left(\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla) \vec{v} \right) = \vec{k} - \nabla p + \eta \Delta \vec{v}$$

where ρ is the density of the fluid, \vec{v} is the velocity vector, \vec{k} are the volume forces, p is the pressure field and η is the dynamic viscosity of the fluid. Here ∇ is the nabla operator and Δ is the Laplace operator.

The term $\frac{\partial \vec{v}}{\partial t}$ describes the temporal change of the velocity, the term $(\vec{v} \cdot \nabla) \vec{v}$ describes the momentum changes due to convection, the term ∇p describes changes of momentum due to gradients in the pressure field and the term $\eta \Delta \vec{v}$ is the momentum diffusion due to the viscous properties of the fluid.

In microfluidic flows the Reynolds numbers are usually small which allows dropping the convection term from the equation. Furthermore if the stationary flow case is considered (i.e., there are no changes of the velocity over time), the time-dependent term can be dropped as well. Finally, if there are no volume forces (gravity, electrokinetic forces, etc.) acting, the equation can be simplified to

$$\Delta \vec{v} = \frac{1}{\eta} \nabla p$$

In most microfluidic applications, the flow is parallel and driven by a pressure drop along the length of the channel i.e.,

along the x -axis (see Fig. 1). There are no y - and z -components of the velocity which simplifies the equation to

$$\Delta v_x = \frac{1}{\eta} \frac{\partial p}{\partial x} = \frac{1}{\eta} \frac{dp}{dx}$$

which amounts to

$$\frac{\partial^2 v_x}{\partial x^2} + \frac{\partial^2 v_x}{\partial y^2} + \frac{\partial^2 v_x}{\partial z^2} = \frac{1}{\eta} \frac{dp}{dx}$$

If we assume the flow profile to be fully developed along the channel axis, the change along x will be zero which simplifies the equation to

$$\begin{aligned} \frac{\partial^2 v_x}{\partial y^2} + \frac{\partial^2 v_x}{\partial z^2} &= \frac{1}{\eta} \frac{dp}{dx} \\ \frac{\partial^2 v_x}{\partial y^2} + \frac{\partial^2 v_x}{\partial z^2} &= \frac{1}{\eta} \frac{\Delta p}{\Delta L} \end{aligned} \quad (1)$$

where we assume a pressure drop Δp along the channel length ΔL in the direction of the x -axis. As we can see, the right-hand side of Eq. 1 is a constant. Eq. 1 is a so-called Poisson equation and it is the Navier–Stokes equation in the form used for parallel fully-developed stationary laminar flows. It is a second-order partial inhomogeneous differential equation. It is this equation which needs to be solved in order to obtain the velocity profile. For simple flow channel geometries (rectangular, circular, etc.) analytical solutions exist. However, once the channel cross-section becomes more complicated, one has to default to numerical methods for solving this equation.

2.2 Finite difference schemes

In the following we will use a so-called finite difference scheme to solve Eq. 1. Finite difference schemes seek solutions to a differential equation on a finite number of intervals using the values of neighbouring intervals $i+1$, $i-1$, $i+2$, $i-2$, etc. to calculate the value of the function f in the interval i . To be more precise these schemes use the difference between the current interval i and the neighbouring intervals which is where the term “difference” stems from. Equation 1

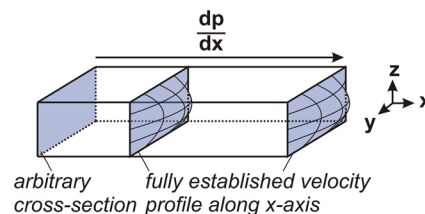


Fig. 1 Schematic of an arbitrary microfluidic flow channel geometry with the coordinate axes. The flow is parallel i.e., there is only the velocity v_x -component to consider. The velocity profile is fully established along the x -axis i.e., v_x is only a function of y and z but not of x

tells us that the change in the first partial derivative $\partial v_x / \partial y$ of v_x with respect to y plus changes of the first partial derivative $\partial v_x / \partial y$ of v_x with respect to z must equal to a constant i.e., the right-hand side of Eq. 1.

We will now derive a numerical scheme for Eq. 1 using the Taylor series.

2.3 Taylor series

The Taylor series states that we can approximate any function $f(x)$ around a given point x_0 if we know the derivatives of this function. The function is thus “expanded” around the location x_0 . The function is then approximated using

$$f(x) = \sum_{n=0}^{n_{\max}} \frac{1}{n!} \frac{d^n f}{dx^n} \bigg|_{x_0} (x-x_0)^n + O_{n_{\max}+1}$$

where n_{\max} is referred to as the “expansion order” and $O_{n_{\max}}$ is the error function of order n_{\max} which sums up all terms of higher order which are omitted. Using this equation we can now approximate the value of the function $f(x)$ when moving in the positive x -direction by Δx to be

$$f(x_0 + \Delta x) = \sum_{n=0}^{n_{\max}} \frac{1}{n!} \frac{d^n f}{dx^n} \bigg|_{x_0} (x_0 + \Delta x - x_0)^n + O_{n_{\max}+1}$$

$$f(x_0 + \Delta x) = \sum_{n=0}^{n_{\max}} \frac{1}{n!} \frac{d^n f}{dx^n} \bigg|_{x_0} \Delta x^n + O_{n_{\max}+1}$$

whereas the change in the negative x -direction is given by

$$f(x_0 - \Delta x) = \sum_{n=0}^{n_{\max}} \frac{1}{n!} \frac{d^n f}{dx^n} \bigg|_{x_0} (x_0 - \Delta x - x_0)^n + O_{n_{\max}+1}$$

$$= \sum_{n=0}^{n_{\max}} \frac{1}{n!} \frac{d^n f}{dx^n} \bigg|_{x_0} (-\Delta x)^n + O_{n_{\max}+1}$$

$$f(x_0 - \Delta x) = \sum_{n=0}^{n_{\max}} (-1)^n \frac{1}{n!} \frac{d^n f}{dx^n} \bigg|_{x_0} \Delta x^n + O_{n_{\max}+1}$$

Expanding both equations to only $n_{\max}=2$ yields

$$f(x_0 + \Delta x) = f(x_0) + \frac{df}{dx}(x_0)\Delta x + \frac{1}{2} \frac{d^2 f}{dx^2}(x_0)(\Delta x)^2 + O_3$$

and

$$f(x_0 - \Delta x) = f(x_0) - \frac{df}{dx}(x_0)\Delta x + \frac{1}{2} \frac{d^2 f}{dx^2}(x_0)(\Delta x)^2 + O_3$$

Adding the last two equations results in

$$f(x_0 + \Delta x) + f(x_0 - \Delta x) = 2f(x_0) + \frac{d^2 f}{dx^2}(x_0)(\Delta x)^2$$

which we can solve for

$$\frac{d^2 f}{dx^2}(x_0) = \frac{f(x_0 + \Delta x) + f(x_0 - \Delta x) - 2f(x_0)}{(\Delta x)^2} \quad (2)$$

We now have a numerical approximation of the second derivative of the function f . As can be seen, we require the value of the function f at x_0 as well as at the positions $x_0 + \Delta x$ and $x_0 - \Delta x$ to derive it. Please note that this approximation is also valid for partial differentials.

2.4 Numerical scheme

We can now use Eq. 2 to approximate the second-order partial differentials in Eq. 1 numerically which yields

$$\begin{aligned} & \frac{f(y_0 + \Delta y) + f(y_0 - \Delta y) - 2f(y_0)}{(\Delta y)^2} \\ & + \frac{f(z_0 + \Delta z) + f(z_0 - \Delta z) - 2f(z_0)}{(\Delta z)^2} \\ & = \frac{1}{\eta} \frac{\Delta p}{\Delta L} \end{aligned} \quad (3)$$

where we introduce the notation $F^{(y,z)}$ for the value of f at the position (y_0, z_0) , $F^{(y+1,z)}$ for the value of f at the position $(y_0 + \Delta y, z_0)$, $F^{(y-1,z)}$ for the value of f at the position $(y_0 - \Delta y, z_0)$, $F^{(y,z+1)}$ for the value of f at the position $(y_0, z_0 + \Delta z)$ and $F^{(y,z-1)}$ for the value of f at the position $(y_0, z_0 - \Delta z)$. We also assume that the so-called step width is equal along the y - and the z -direction, i.e., $h = \Delta y = \Delta z$ which allows us to write Eq. 3 as

$$\frac{F^{(y+1,z)} + F^{(y-1,z)} + F^{(y,z+1)} + F^{(y,z-1)} - 4F^{(y,z)}}{h^2} = \frac{1}{\eta} \frac{\Delta p}{\Delta L}$$

which we solve for $F^{(y,z)}$ finding

$$F^{(y,z)} = \frac{1}{4} \left(F^{(y+1,z)} + F^{(y-1,z)} + F^{(y,z+1)} + F^{(y,z-1)} - \frac{h^2}{\eta} \frac{\Delta p}{\Delta L} \right) \quad (4)$$

Equation 4 is the numerical scheme which allows us to solve Eq. 1. However, in order to solve for the (unknown) value $F^{(y,z)}$ we require the (also unknown) values $F^{(y+1,z)}$, $F^{(y-1,z)}$, $F^{(y,z+1)}$ and $F^{(y,z-1)}$. This is not problematic for numerical methods which often require unknown values of neighbouring intervals, cells (in FVM) or elements (in FEM) to calculate the value of the current interval (volume or cell). This is usually tackled by an iterative approach whereby a first solution is generated with the (estimated) neighbouring values whereupon the value of the current interval can be calculated. This value is then used to update the values of the neighbouring cells resulting in a better approximation of the cells values. Once these have been determined, the value of the current interval can be adapted thereby obtaining an even better solution and so on. This process is repeated until the

solution is sufficiently exact, i.e., until the changes per iteration are smaller than a (user-defined) threshold value.

2.5 A note on numerical stability

Stability considerations are important whenever numerical methods are used to solve (partial) differential equations. It is beyond the scope of this article to discuss the exact criteria of numerical stability. As a short summary one can state that the higher the order of the numerical scheme, i.e., the higher the order of the error function, the more stable a numerical scheme will be. If a numerical scheme becomes instable a solver will usually abort because the values obtained from it become physically meaningless. Equation 4 was derived from the numerical schemes for the second-order (partial) differentials of Eq. 2. The latter was a numerical scheme of order 2, i.e., with an error function of order 3. As a consequence, Eq. 4 has a decent numerical stability and instability is not an issue.

3 Implementation

3.1 Units

Before we implement the numerical scheme of Eq. 4 we will make sure that we obtain correct values. The velocity values $F^{(y,z)}$, $F^{(y+1,z)}$, $F^{(y-1,z)}$, $F^{(y,z+1)}$ and $F^{(y,z-1)}$ will be given in mm/s. The pressure drop $\Delta p/\Delta L$ will be given in mbar/mm, the dynamic viscosity η in mPa·s. The step width h will be given in μm . We sum up the term originating from the right-hand side of Eq. 1 as

$$\Gamma = 0.1 \frac{h^2}{\eta} \frac{\Delta p}{\Delta L} \quad (5)$$

The prefactor has been added to correct the unit to mm/s.

3.2 Step 1: Setting up the spreadsheet

We start by creating a new spreadsheet in Microsoft Excel. The first step is shown in Fig. 2a. In the [supplementary material](#) you will find a ready-made spreadsheet created using Microsoft Excel 2010 which you can directly use. First we create a square microfluidic channel. For this, we adjust a total of 40×40 cells on the left-hand side of the spreadsheet (cells B2 to AO41). These cells will be the domain, i.e., the microfluidic channel for which we want to determine the velocity profile. Adjust the column width such that the cells are nearly square in shape. Leave a one-cell border around this area. This border represents the walls of the microfluidic channel. For better visualization you may want to colour these cells in grey. Insert

the value '0' into these cells as the velocity at the channel wall is 0 (no-slip boundary condition).

Next to the channel, we create a colour scale. For this we determine the minimum and the maximum value of the velocity in our domain. For obtaining the maximum value we select the cell and insert

$$= \text{MAX}(A1 : AP42)$$

as the function. This will query the maximum velocity value in our domain which is stored in cell AR16. Similarly type

$$= \text{MIN}(A1 : AP42)$$

for obtaining the minimum velocity value in our domain which is stored in cell AR18. We create the colour scale by copying the minimum value into cell AR13 and adding a total of 9 cells above this cell each of which will be incremented by the value of $(\text{MAX}-\text{MIN})/9$ by inserting the formula

$$= \text{AR}\{\text{line below}\} + (\text{AR}\$16-\text{AR}\$18)/9$$

By this way we create a block of cells containing increasing numbers from AR13 to AR4. In order to see this effect we arbitrarily set one of the cells in the domain to the value of 5 (see Fig. 2a). In the next step, we simply copy these cells into the cells AU14 to AU4. The latter cells will be turned into a colour bar (see step 2).

We also add a number of variables which can be changed by the user. For this we create an entry for the pressure drop (in mbar/mm, here entered in cell AW21 and set to -0.1 mbar/mm), the step width (in μm , here entered in cell AW22 and set to 2.5 μm), the dynamic viscosity (in mPa·s, here entered in cell AW 23 and set to 1 mPa·s) and the Value Γ calculated from these values according to Eq. 5 (here entered in cell AW24 and calculated as -0.0625 mm/s). As stated, these variables can be changed by the user.

3.3 Step 2: Colouring

Whenever we see the result of numerical calculations we are used to seeing the numerical data not in form of numbers but rather as false-colour plots. The most commonly employed colouring scheme is 'red-yellow-green'. We first select the cells of the domain (cells B2 to AO41) then select 'conditional formatting- > colour scales' and select the second entry 'red-yellow-green'. As you can see, the cell with the arbitrarily set value of '5' will turn red. By customizing the formatting we can remove the values which will give an even better representation. For this we right-click and select 'format cells- > number- > custom' and enter ';;;'. This removes the cells values. As a consequence, the number '5' disappears and we are left with a single red cell. We do the

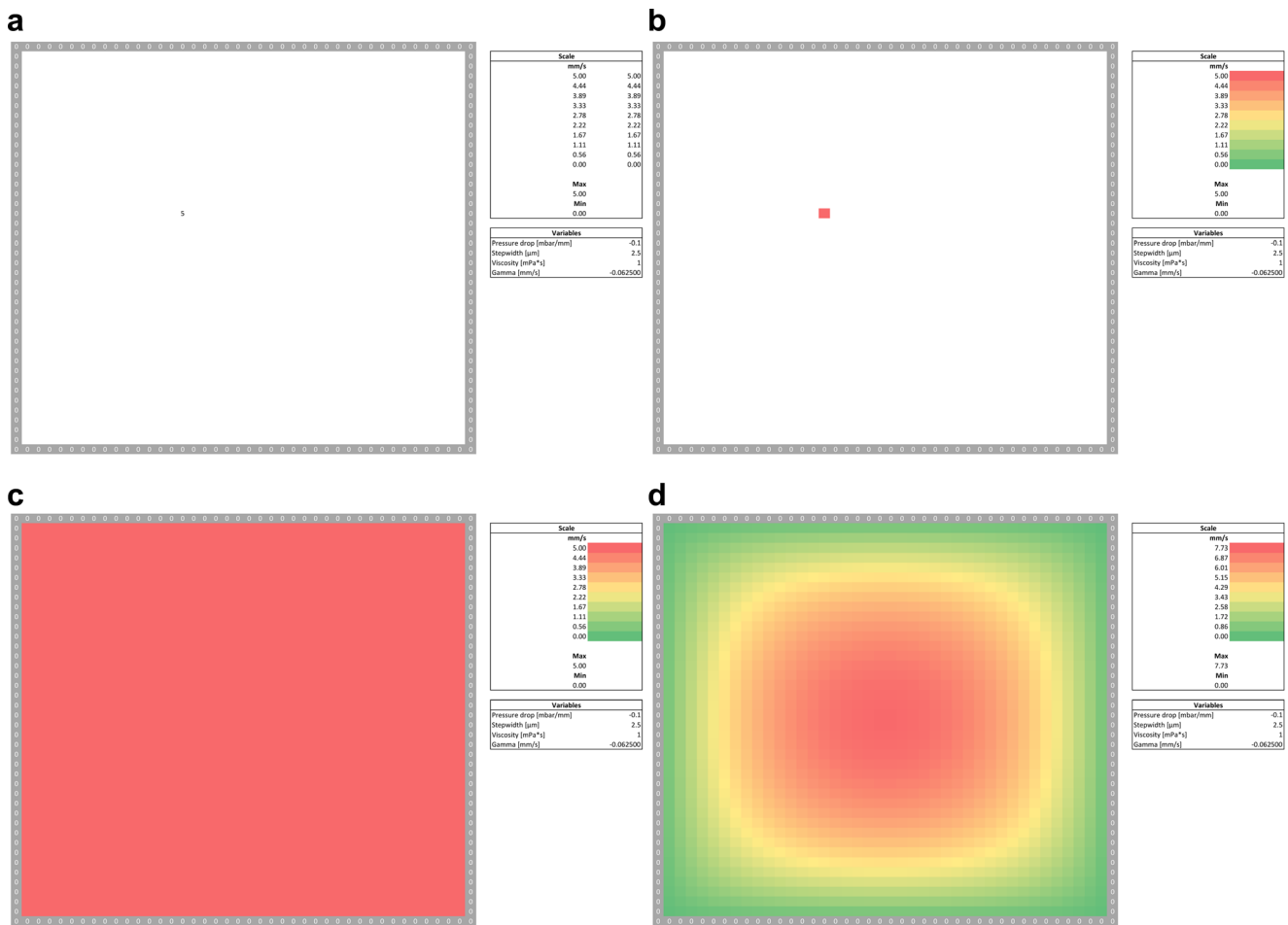


Fig. 2 Development of the Microsoft Excel spreadsheet for solving the Navier–Stokes equation for Poiseuille flow numerically using a FDM scheme. **a** Spreadsheet after setting up the domain with the boundary (in grey), the colour scale and the variables section (step 1). **b** Spreadsheet after activating colour-coding (step 2). **c** Spreadsheet after implementing the numerical scheme of Eq. 4 (step 3). **d** Spreadsheet after

activating iterative calculations (step 4). As can be seen the output shows the expected parabolic-shaped velocity profile. The channel was assumed to be 100 μm wide and 100 μm high. The applied pressure drop was −0.1 mbar/mm and the viscosity 1 mPa·s. The step width h of the grid was 2.5 μm. A total of 40 × 40 cells were used

same with cells AU14 to AU4 which will yield a colour scale. Figure 2b shows the spreadsheet after these modifications.

3.4 Step 3: Implementing the numerical scheme

In the third step we now implement the numerical scheme. For this we select the top-left cell in the domain (cell B2) and enter the following formula

$$= 0.25*(B1 + A2 + B3 + C2-\$AU\$24)$$

This implements Eq. 4. The value at the position AU24 is Γ . As you will see, the cell turns green. We then copy this value from the first cell into all the cells of our domain by simply selecting it and dragging the bottom-right edge of the cell across the domain. Microsoft Excel will

most likely issue an error stating that a circular reference has been created. This error can be ignored for the moment. After copying the cells, the whole domain will be coloured in red. The output is shown in Fig. 2c.

3.5 Step 4: Activating iteration

Obviously this result is not correct. As already stated, Eq. 4 is an iterative scheme and the spreadsheet tools must be configured to perform iterative calculations. As mentioned, Microsoft Excel will most likely have issued an error stating that a circular reference has been created. In circular references the values of two cells are interdependent. As a consequence, Microsoft Excel will need to update the value of the first cell after having updated the value of the second cell whereupon the value of the first cell must be revaluated and so on. As this creates infinite

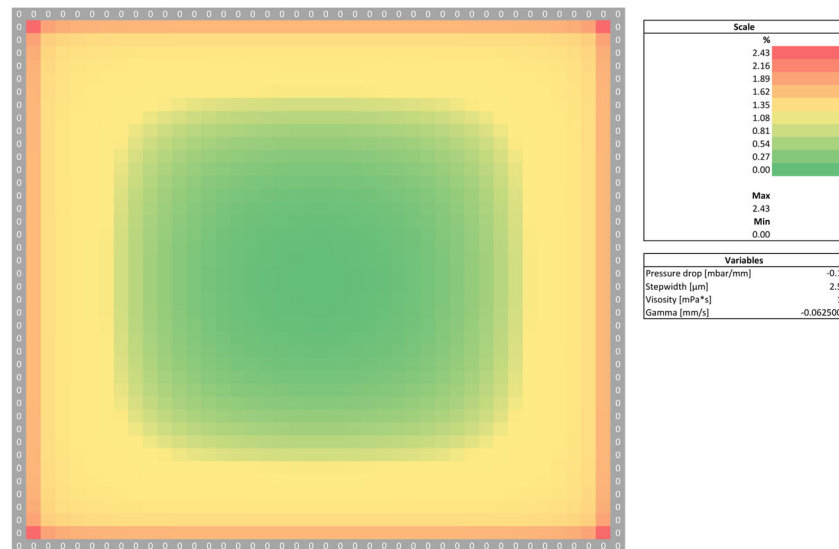


Fig. 3 Accuracy of the numerically obtained velocity profile from Fig. 2 compared to the analytical solution given by Eq. 6 expanded to $n_{\max} = 50$. The plot shows the absolute error in percent. As can be seen, the

maximum error is around 2.5 % and occurs in the corners of the profile where the relative changes in the dependent variable are highest

loops, spreadsheet tools will not perform iterative calculations. However, Microsoft Excel can be configured to carry out a finite number of iterations thereby allowing iterative calculations. For this select 'File- > Options- > Formulas' and check 'Enable iterative calculation'. The number of required iterations can be set to a value of about 1000 which is sufficient in most cases. We can also apply a second condition stating that iterations should be continued until a given threshold value has been reached, i.e., the value of a cell changes by an amount smaller than this value per iteration. Usually a reasonable value is around 10^{-5} . After a short moment, the domain will look as shown in Fig. 2d. As can be seen, this resembles the parabolic velocity profile which is expected for Poiseuille flow in rectangular channels.

4 Results and discussion

4.1 Exactness of the solution

We have now setup a spreadsheet that solves the Navier-Stokes equation for Poiseuille flow Eq. 1 on a square channel cross-section. As the chosen step width h was $2.5 \mu\text{m}$ the length and width of the channel was $100 \mu\text{m}$ (as we used a total of 40×40 cells). The displayed colour-coded velocity profile is calculated using the iterative numerical Eq. 4 which Microsoft Excel solves for us within a few seconds. Obviously it remains to check how exact the solution is. For this, we compare the velocity profile obtained with the profile calculated from the analytical solution to Eq. 1. We will not derive this solution here, it is given by

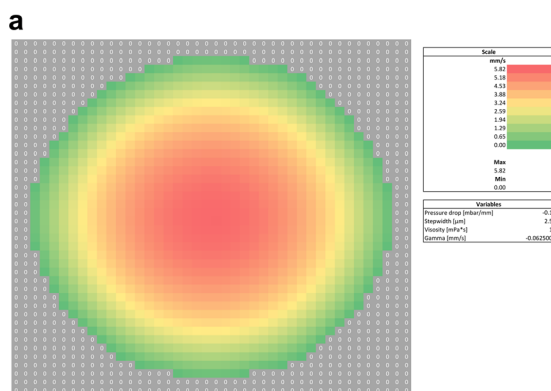
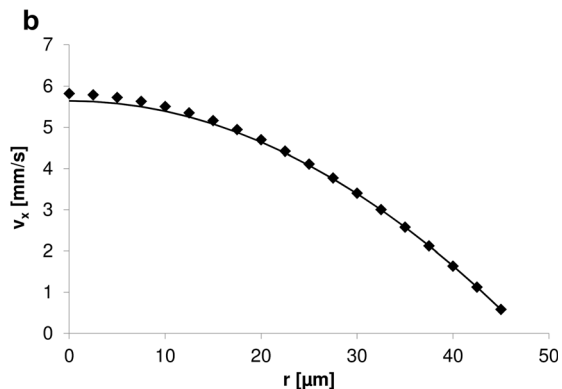


Fig. 4 Usage of the developed spreadsheet for obtaining the well-known parabolic flow profile of the Hagen-Poiseuille flow in circular channels. **a** Output of the spreadsheet. A circular cross-section of $95 \mu\text{m}$ diameter was assumed. The domain was again divided into 40×40 cells. The applied



pressure drop was -0.1 mbar/mm and the viscosity $1 \text{ mPa}\cdot\text{s}$. The step width h of the grid was $2.5 \mu\text{m}$. **b** Comparison of the numerically obtained solution (diamonds) with the analytical solution (solid line) calculated using Eq. 7. As can be seen the results are in good agreement

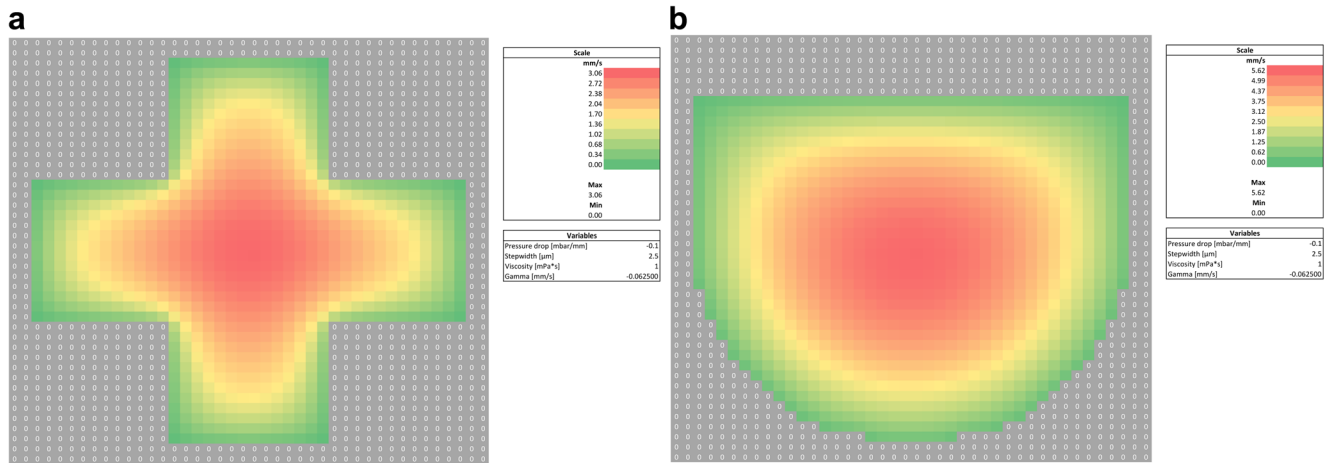


Fig. 5 Usage of the spreadsheet to solve the Navier–Stokes equation for Poiseuille flow on more complex channel cross-sections: **a** cross-shaped channel, **b** extended semi-circular channel. In all cases, a step width h of 2.5 μm , an applied pressure drop of -0.1 mbar/mm and a viscosity 1

$$v_x(y, z) = -\frac{4h^2}{\eta\pi^3} \frac{dp}{dx} \sum_{n=0}^{\infty} \frac{1}{(2n+1)^3} \left(1 - \frac{\cosh\left((2n+1)\pi\frac{y}{h}\right)}{\cosh\left((2n+1)\pi\frac{w}{2h}\right)} \right) \cdot \sin\left(\pi\frac{z}{h}(2n+1)\right) \quad (6)$$

where h is the height of the channel (in μm), and w is the width of the channel (in μm). Figure 3 shows the absolute error of our numerical solution compared to the analytical solution. As can be seen, the error is highest in regions with the highest gradients. The maximum error encountered is in the range of around 2.5 % which is sufficiently exact for most applications.

4.2 Hagen-Poiseuille flow

Now that we have the spreadsheet available, we are able to use it for different channel geometries. Figure 4a shows the well-known parabolic flow profile of the Hagen-Poiseuille flow derived for a circular cross-section. Again, we will not derive the analytical solution which is given by

$$v_x(r) = -\frac{R^2}{4\eta} \frac{dp}{dx} \left(\left(\frac{r}{R} \right)^2 - 1 \right) \quad (7)$$

In this example, we assume a pressure drop of -0.1 mbar/mm and a viscosity of 1 mPa·s. The step width h of the grid was 2.5 μm and a circular-shaped capillary with a diameter of 95 μm was assumed. The output of the spreadsheet is shown in Fig. 4a. As you can see, the numerically obtained profile is parabolic as well. Figure 4b shows the comparison of the numerical with the analytical solution given by Eq. 7. As

mPa·s. were used. Please note that these cross-sections would be very difficult to tackle analytically whereas the spreadsheet recovers the solution almost instantaneously

can be seen, the numerical solution is very close to the analytical solution.

4.3 More complex channel cross sections

Obviously it is not necessary to derive the Hagen-Poiseuille flow profile numerically as this highly-symmetrical profile is best tackled analytically. However, as can be seen, the developed spreadsheet lends itself readily to solving a wide variety of cross-sections. Figure 5 shows just two examples of very complex microfluidic channel cross-sections which would be difficult to solve analytically. The shape shown in Fig. 5b is commonly encountered in microfluidic channel structures made by thermoforming. One example of such a structure is shown in Fig. 6. However, once the respective channel boundaries are set in the spreadsheet, the flow profiles are calculated within seconds. This shows how easy and convenient numerical methods can be applied by using readily available tools such as Microsoft Excel.

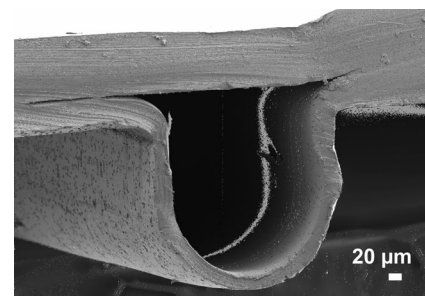


Fig. 6 Example of a microfluidic channel with an extended semi-circular cross-section corresponding to the profile shown in Fig. 5b. This channel was created by means of microthermoforming (Truckenmüller et al. 2011). The material of the channel is heavy-ion irradiated track etched polycarbonate (thickness 65 μm , it4ip, Belgium)

5 Conclusions

In this paper we have demonstrated that the Navier–Stokes equation for Poiseuille flow can be readily solved numerically using conveniently accessible spreadsheet analysis tool such as Microsoft Excel. For this, a numerical scheme based on a finite different method was implemented. The spreadsheet can be quickly adapted to any channel cross-section for which it will derive the velocity profile within the microfluidic channel almost instantaneously. Numerical methods have become an essential part in the design and optimization of microfluidic systems. However, many researchers inexperienced with numerical solver packages often are not trained in correctly applying the often very complex software packages. Additionally, licensing of these packages is often expensive and many researchers do not have access to suitable academic licences. In many applications, researchers may want to derive the velocity profile in a non-standard microfluidic flow channel geometry in order to assess the properties of the flow inside of this channel. Many important criteria (such as, e.g., the shear stress profiles) are directly derivable from the velocity profile. Being able to quickly and conveniently derive these velocity profiles by solving the Navier–Stokes equation

numerically using a widely available tool such as Microsoft Excel will allow a significantly wider audience of researchers to use the advantages of numerical methods.

Acknowledgments This work was funded by the Bundesministerium für Bildung und Forschung (BMBF), funding code 03X5527 “Fluoropor”

References

- S. Khodaparast, N. Borhani, J.R. Thome, Sudden expansions in circular microchannels: flow dynamics and pressure drop. *Microfluid. Nanofluid.* **17**, 561–572 (2014). doi:[10.1007/s10404-013-1321-7](https://doi.org/10.1007/s10404-013-1321-7)
- J. Kim, J. Lee, C. Wu, S. Nam, D. Di Carlo, W. Lee, Inertial focusing in non-rectangular cross-section microchannels and manipulation of accessible focusing positions. *Lab Chip* **16**, 992–1001 (2016)
- R. Truckenmüller et al., Thermoforming of film-based biomedical microdevices. *Adv. Mater.* **23**, 1311–1329 (2011)
- A. Waldbaur, B. Carneiro, P. Hettich, E. Wilhelm, B. Rapp, Computer-aided microfluidics (CAMF): from digital 3D-CAD models to physical structures within a day. *Microfluid. Nanofluid.* **15**, 625–635 (2013). doi:[10.1007/s10404-013-1177-x](https://doi.org/10.1007/s10404-013-1177-x)
- E. Wilhelm, C. Neumann, T. Duttendorfer, L. Pires, B.E. Rapp, Connecting microfluidic chips using a chemically inert, reversible, multichannel chip-to-world-interface. *Lab Chip* **13**, 4343–4351 (2013). doi:[10.1039/c3lc50861g](https://doi.org/10.1039/c3lc50861g)